

TITLE OF THE INVENTION

DECODING APPARATUS AND DECODING METHOD

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is based upon and claims the  
5 benefit of priority from the prior Japanese Patent  
Applications No. 2000-092163, filed March 29, 2000; and  
No. 2001-071358, filed March 14, 2001, the entire  
contents of both of which are incorporated herein by  
reference.

10 BACKGROUND OF THE INVENTION

The present invention relates to Reed-Muller  
decoding apparatus and decoding method.

Reed-Muller code is known as a kind of error  
correction code. An ordinary Reed-Muller code is (32,  
15 6) Reed-Muller code for converting 6-bit information  
data into a 32-bit code word. For the Reed-Muller  
code, it is known that, suppose  $n = 2^m$  ( $n$  is a code  
length,  $m$  is a natural number (if  $n = 32$ ,  $m = 5$ )), the  
minimum Euclidean distance between code words is  $2^{m-r}$   
20 ( $r$  is an order of code). In general, if the minimum  
Euclidean distance between code words is longer, the  
error correction code has better performance (resistant  
to errors). However, the longer the minimum Euclidean  
distance, the lower the transmission rate or coding  
25 efficiency. Therefore, in order to improve the  
performance of the Reed-Muller code without greatly  
lowering the transmission rate, a method is proposed

to increase the minimum Euclidean distance by adding  
mask symbols to the conventional Reed-Muller code (3rd  
Generation Partnership Project; Technical Specification  
Group Radio Access Network; Multiplexing and channel  
coding (FDD) (Release 1999) 3G TS 25.212 V3.3.0  
5 (2000-06). This code is called "(32, 10) Reed-Muller  
code" for converting a total 10-bit information data  
where 4-bit mask symbols are added to a 6-bit  
information data into a 32-bit code word.

10 It is known that the Reed-Muller code decoding  
apparatus can be realized by a simple majority  
decision circuit (Jpn. Pat. Appln. KOKAI Publication  
No. 9-74359). The majority decision circuit for (32,  
6) Reed-Muller code can be realized relatively easily.  
15 However, for (32, 10) Reed-Muller code, it is difficult  
to calculate the checksum to be determined for the  
majority decision.

As an example of decoding without using a majority  
decision circuit, a maximum likelihood decoding by  
20 calculating a correlation value is known (Harmonization  
impact of TFCI and New Optimal Coding for extended TFCI  
with almost no complexity increase (rev 1) TSGR #6 (99)  
970). However, calculating the correlation of all  
code words for a received coded signal, essentially,  
25 operation load is high in this method, increasing the  
hardware scale; therefore, this method is difficult to  
realize for (32, 10) Reed-Muller code.

As mentioned above, it has been difficult to realize the decoding apparatus for recently proposed Reed-Muller code containing mask symbols which is resistant to the error.

5

#### BRIEF SUMMARY OF THE INVENTION

Accordingly, the present invention is directed to method and apparatus that substantially obviates one or more of the problems due to limitations and disadvantages of the related art.

10

In accordance with the purpose of the invention, as embodied and broadly described, the invention is directed to reduce the operation load and hardware scale of the decoding apparatus using mask symbols.

15

According to the present invention, there is provided an apparatus for decoding Reed-Muller code in which information data is encoded by using mask symbols and orthogonal codes, the information data including a first portion and a second portion, the apparatus comprising:

20

an arithmetic operation unit configured to calculate a first exclusive OR of the Reed-Muller code and an exclusive ORed value of a candidate pattern of the mask symbols and the information data corresponding to the candidate pattern;

25

a first decoder configured to calculate a checksum of the first exclusive OR and majority-decide the checksum to decode a part of the second portion of the

information data corresponding to the orthogonal codes;

a second decoder configured to calculate a second exclusive OR of the first exclusive OR and a product of the part of the second portion of the information data and the orthogonal codes and majority-decide the second exclusive OR to decode a remaining part of the second portion of the information data corresponding to the orthogonal codes;

10 a Reed-Muller encoder configured to Reed-Muller encode the second portion of the information data output from the first decoder and the second decoder and the first portion of the information data;

15 a minimum distance detector configured to detect the minimum of a Euclidean distance between an output from the Reed-Muller encoder and the Reed-Muller code supplied to the arithmetic operation unit while a plurality of candidate patterns of the mask symbols are supplied to the arithmetic operation unit,

20 whereby the first portion of the information data is decoded based on the mask symbols corresponding to the minimum of the Euclidean distance.

According to the present invention, there is provided a method of decoding Reed-Muller code in which information data is encoded by using mask symbols and orthogonal codes, the information data including a first portion and a second portion, the method comprising:

DRAFT EDITION 2000

calculating a first exclusive OR of the Reed-Muller code and an exclusive ORed value of a candidate pattern of the mask symbols and the information data corresponding to the candidate pattern;

5 calculating a checksum of the first exclusive OR and majority-judging the checksum to decode a part of the second portion of the information data corresponding to the orthogonal codes;

10 calculating a second exclusive OR of the first exclusive OR and a product of the part of the second portion of the information data and the orthogonal codes and majority-judging the second exclusive OR to decode a remaining part of the second portion of the information data corresponding to the orthogonal codes;

15 Reed-Muller encoding the decoded second portion of the information data and the first portion of the information data; and

20 detecting the minimum of a Euclidean distance between the Reed-Muller encoded data and an input Reed-Muller code while a plurality of first exclusive ORs are calculated, whereby the first portion of the information data is decoded based on the mask symbols corresponding to the minimum of the Euclidean distance.

25 According to the present invention, there is provided another apparatus for decoding Reed-Muller code in which information data is encoded by using mask symbols and orthogonal codes, the information data

including a first portion and a second portion, the apparatus comprising:

an arithmetic operation unit configured to calculate a first product of the Reed-Muller code and  
5 an exclusive ORed value of a candidate pattern of the mask symbols and the information data corresponding to the candidate pattern;

a first decoder configured to calculate a checksum of the first product and majority-decide the checksum  
10 to decode a part of the second portion of the information data corresponding to the orthogonal codes;

a second decoder configured to calculate a second product of the first product and a product of the part of the second portion of the information data and the  
15 orthogonal codes and majority-decides the second product to decode a remaining part of the second portion of the information data corresponding to the orthogonal codes;

a Reed-Muller encoder configured to Reed-Muller encode the second portion of the information data output from the first decoder and the second decoder  
20 and the first portion of the information data;

a maximum correlation detector configured to detect the maximum of a correlation between an output  
25 from the Reed-Muller encoder and the Reed-Muller code supplied to the arithmetic operation unit while a plurality of candidate patterns of the mask symbols

are supplied to the arithmetic operation unit,  
whereby the first portion of the information data  
is decoded based on the mask symbols corresponding to  
the maximum of the correlation.

5        According to the present invention, there is  
provided another method of decoding Reed-Muller code in  
which information data is encoded by using mask symbols  
and orthogonal codes, the information data including  
a first portion and a second portion, the method  
10      comprising:

calculating a first product of the Reed-Muller  
code and an exclusive ORed value of a candidate  
pattern of the mask symbols and the information data  
corresponding to the candidate pattern;

15      calculating a checksum of the first product and  
majority-decide the checksum to decode a part of the  
second portion of the information data corresponding to  
the orthogonal codes;

20      calculating a second product of the first product  
and a product of the part of the second portion of the  
information data and the orthogonal codes and majority-  
decides the second product to decode a remaining part  
of the second portion of the information data  
corresponding to the orthogonal codes;

25      Reed-Muller encoding the second portion of  
the information data and the first portion of the  
information data;

detecting the maximum of a correlation between the Reed-Muller encoded data and an input Reed-Muller code while a plurality of first products are calculated, whereby the first portion of the information data is  
5 decoded based on the mask symbols corresponding to the maximum of the correlation.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

FIG. 1 shows the first embodiment of the decoding apparatus according to the present invention;

10 FIG. 2 is a flow chart showing an operation flow of the first embodiment;

FIG. 3 shows a modification of the checksum calculator and the majority decision circuit of the first embodiment;

15 FIG. 4 shows the second embodiment of the decoding apparatus according to the present invention;

FIG. 5 is a flow chart showing an operation flow of the second embodiment;

20 FIG. 6 shows the third embodiment of the decoding apparatus according to the present invention;

FIG. 7 is a flow chart showing an operation flow of the third embodiment;

25 FIG. 8 shows a modification of the checksum calculator and the majority decision circuit of the third embodiment;

FIG. 9 shows the fourth embodiment of the decoding apparatus according to the present invention; and

FIG. 10 is a flow chart showing an operation flow of the fourth embodiment.

#### DETAILED DESCRIPTION OF THE INVENTION

An embodiment of a decoding apparatus according 5 to the present invention will now be described with reference to the accompanying drawings.

##### First embodiment

FIG. 1 shows a decoding apparatus of (32, 10) Reed-Muller code according to the first embodiment of 10 the present invention. In (32, 10) Reed-Muller code, as the mask symbols are selected by 4-bit information data, patterns of the mask symbols (mask patterns) are  $2^4 = 16$  patterns in total.

The following definition will be used for the 15 description below.

" $\wedge$ " means an exclusive OR operation. For two vectors, A and B, " $A \wedge B$ " represents the exclusive OR of components of respective vectors A and B.

$m(A)$  represents the vector A in which each of 20 components 0 and 1 is changed to +1 and -1.

10-bit information data to be encoded are assumed to be  $d_0, d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8$  and  $d_9$ . Each bit data  $d_n$  is 0 or 1.

Orthogonal codes used for encoding are assumed to 25 be  $C_0, C_1, C_2, C_3, C_4$  and  $C_5$ . Each code  $C_n$  is a 32-bit data, and 32 elements thereof are 0 or 1. Note that  $C_0$  is a series of all 1.

Similarly, assuming mask symbols used for encoding be  $M_1$ ,  $M_2$ ,  $M_3$  and  $M_4$ . Each mask symbol  $M_n$  is a 32-bit data. The mask patterns  $d_6M_1 \wedge d_7M_2 \wedge d_8M_3 \wedge d_9M_4$ , which are exclusive ORs of the mask symbols and the information data, have  $2^4 = 16$  patterns.

5

Examples of the orthogonal codes  $C_0$  to  $C_5$  and the mask symbols  $M_1$  to  $M_4$  are shown in Table 1.

Table 1

i	$C_{i,0}$	$C_{i,1}$	$C_{i,2}$	$C_{i,3}$	$C_{i,4}$	$C_{i,5}$	$M_{i,1}$	$M_{i,2}$	$M_{i,3}$	$M_{i,4}$
0	1	0	0	0	0	1	0	0	0	0
1	1	0	0	0	1	0	1	0	0	0
2	1	0	0	0	1	1	0	0	0	1
3	1	0	0	1	0	0	1	0	1	1
4	1	0	0	1	0	1	0	0	0	1
5	1	0	0	1	1	0	0	0	1	0
6	1	0	1	1	1	1	0	1	0	0
7	1	0	1	0	0	0	0	1	1	0
8	1	0	1	0	0	1	1	1	1	0
9	1	0	1	0	1	0	1	0	1	1
10	1	0	1	0	1	1	0	0	1	1
11	1	0	1	1	0	0	0	1	1	0
12	1	0	1	1	0	1	0	1	0	1
13	1	0	1	1	1	0	1	0	0	1
14	1	0	1	1	1	1	1	1	1	1
15	1	1	0	0	0	1	1	1	0	0
16	1	1	0	0	1	0	1	1	0	1
17	1	1	0	0	1	1	1	0	1	0
18	1	1	0	1	0	0	0	1	1	1
19	1	1	0	1	0	1	0	1	0	1
20	1	1	0	1	1	0	0	0	1	1
21	1	1	0	1	1	1	0	1	1	1
22	1	1	1	0	0	0	0	1	0	0
23	1	1	1	0	0	1	1	1	0	1
24	1	1	1	0	1	0	1	0	1	0
25	1	1	1	0	1	1	1	0	0	1
26	1	1	1	1	0	0	0	0	1	0
27	1	1	1	1	0	1	1	1	0	0
28	1	1	1	1	1	0	1	1	1	0
29	1	1	1	1	1	1	1	1	1	1
30	1	0	0	0	0	0	0	0	0	0
31	1	1	0	0	0	0	1	0	0	0

An encoding apparatus encodes the aforementioned information data "d" based on the orthogonal codes  $C_0$  to  $C_5$  and the mask symbols  $M_1$  to  $M_4$ , and outputs the following 32-bit coded signal "s." Here, the  
5 orthogonal codes and the mask symbols to be multiplied with each bit of the information data are predetermined.

$$s = d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge d_6C_6 \wedge d_7C_7 \wedge d_8C_8 \wedge d_9C_9 \quad (1)$$

10 The 32-bit coded signal "s" is modulated and output as  $m(s)$ . In this embodiment, the following signal in which an error due to transfer path or noise is added to the modulated signal  $m(s)$  is input to the decoding apparatus of FIG. 1, and hard-decided by a  
15 hard decision unit 10.

The hard decision unit 10 reproduces the original values 0 and 1, when the value +1 and -1 corresponding to the original values 0 and 1 becomes other values such as 0.2, 1.8, -1.2 or the like due to noise or the  
20 like. Thus, the hard decision unit 10 outputs a sum of the 32-bit coded signal "s" and an error signal "e."

A memory 12 stores the orthogonal codes  $C_0$  to  $C_5$ , the mask symbols  $M_1$  to  $M_4$  of Table 1, and 16 mask patterns  $d_6M_1 \wedge d_7M_2 \wedge d_8M_3 \wedge d_9M_4$  not shown in Table 1.  
25 Here, "i" represents a bit position.

An exclusive OR circuit 14 calculates an exclusive OR of one of the mask patterns stored in the memory 12

and the output from the hard decision circuit 10.

An output from the exclusive OR circuit 14 is supplied to a checksum calculator 16. The calculator 16 calculates 16 checksums for each of 5 bits of  $d_1$  to 5  $d_5$  (80 checksums in total), among 10-bit information data  $d_0$  to  $d_9$ .

A majority decision unit 18 decides a majority of the 80 checksums output from the checksum calculator 16 to decode bits  $d_1'$  to  $d_5'$  corresponding to the 10 orthogonal codes  $C_1$  to  $C_5$ . To be more specific, concerning the checksum, it is decided to be 0 if 0 is majority, and 1 if 1 is majority.

An orthogonal code multiplier 20 multiplies 5-bit data  $d_1'$  to  $d_5'$  by the orthogonal codes.

15 An exclusive OR circuit 22 calculates an exclusive OR of the exclusive OR output from the exclusive OR circuit 14 and the output from the orthogonal code multiplier 20. A majority decision unit 24 decides a majority of the exclusive OR output from the exclusive 20 OR circuit 22 to decode bit  $d_0'$ . To be more specific, concerning the exclusive OR, it is decided to be 0 if 0 is majority, and 1 if 1 is majority. When the bit  $d_0'$  of the information data is determined by the majority decision unit 24, bits  $d_6'$  to  $d_9'$  of the information 25 data can be determined based on the mask pattern used for determining the bit  $d_0'$ .

The operation mentioned above, i.e., exclusive

ORing the Reed-Muller code input to the decoding apparatus and the exclusive OR of the mask pattern and the information data, allows to exclude the mask pattern from the Reed-Muller code. The Reed-Muller code excluding the mask pattern is easily majority-decided. The bit data  $d_0'$  to  $d_9'$  are determined by multiplying the result of the majority-decision by the orthogonal codes. The bit data  $d_0'$  to  $d_9'$  are Reed-Muller encoded by a Reed-Muller encoder 26. The output from the Reed-Muller encoder 26 is supplied to a Euclidean distance calculator 28. The Euclidean distance between the output from the Reed-Muller encoder 26 and the received coded signal output from the hard decision unit 10 is calculated.

The aforementioned processing is performed for all 16 kinds of mask patterns, and the minimum Euclidean distance is detected by a minimum distance detector 30. Bit data  $d_0'$  to  $d_9'$  at the time when the minimum distance is detected are considered to be correct, completing the decoding.

FIG. 2 is a flow chart of the first embodiment.

In step S10, the hard decision unit 10 hard-decides the coded signal. The coded signal input in this decoding apparatus is not the modulated signal  $m(s)$  output from the encoding apparatus, but the following signal in which error "e" due to transfer path or noise is added to  $m(s)$ .

$$d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge d_6C_6 \wedge d_7C_7 \wedge d_8C_8 \\ \wedge d_9C_9 \wedge e \quad (2)$$

In the hard decision, the original values 0 and 1  
are reproduced when the values +1 and -1 corresponding  
5 to the original values 0 and 1 become other values such  
as 0.2, 1.8, -1.2 or the like due to noise or the like.

One mask pattern is specified in step S12, this  
specified mask pattern is read out from the memory 12  
in step S14, and the exclusive OR circuit 14 calculates  
10 in step S16 the exclusive OR of the coded signal output  
from the hard decision unit 10 and the mask pattern.

The memory 12 stores the orthogonal codes  $C_0$  to  
 $C_5$ , mask symbols  $M_1$  to  $M_4$ , and 16 mask patterns  $d_6M_1$   
 $d_7M_2 \wedge d_8M_3 \wedge d_9M_4$  not shown in Table 1. "i" represents  
15 a bit position.

Supposing the mask pattern read out from the  
memory 12 be  $M' = d_6'M_1 \wedge d_7'M_2 \wedge d_8'M_3 \wedge d_9'M_4$ , the  
exclusive OR output from the exclusive OR circuit 14  
will be as follows.

$$20 \quad d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge (d_6 \wedge d_6')M_1 \wedge (d_7 \\ \wedge d_7')M_2 \wedge (d_8 \wedge d_8')M_3 \wedge (d_9 \wedge d_9')M_4 \wedge e \quad (3)$$

In step S18, the checksum calculator 16 calculates  
the checksum of the expression (3) output from the  
exclusive OR circuit 14. Respectively, 16 checksums  
25 are calculated for 5 bits of  $d_1$  to  $d_5$ , in the 10-bit  
information data of  $d_0$  to  $d_9$ .

Checksums for  $d_1$

d<sub>1</sub>' = r<sub>0</sub> × r<sub>30</sub>  
d<sub>1</sub>' = r<sub>1</sub> × r<sub>2</sub>  
d<sub>1</sub>' = r<sub>3</sub> × r<sub>4</sub>  
d<sub>1</sub>' = r<sub>5</sub> × r<sub>6</sub>  
5 d<sub>1</sub>' = r<sub>7</sub> × r<sub>8</sub>  
d<sub>1</sub>' = r<sub>9</sub> × r<sub>10</sub>  
d<sub>1</sub>' = r<sub>11</sub> × r<sub>12</sub>  
d<sub>1</sub>' = r<sub>13</sub> × r<sub>14</sub>  
d<sub>1</sub>' = r<sub>15</sub> × r<sub>31</sub>  
10 d<sub>1</sub>' = r<sub>16</sub> × r<sub>17</sub>  
d<sub>1</sub>' = r<sub>18</sub> × r<sub>19</sub>  
d<sub>1</sub>' = r<sub>20</sub> × r<sub>21</sub>  
d<sub>1</sub>' = r<sub>22</sub> × r<sub>23</sub>  
d<sub>1</sub>' = r<sub>24</sub> × r<sub>25</sub>  
15 d<sub>1</sub>' = r<sub>26</sub> × r<sub>27</sub>  
d<sub>1</sub>' = r<sub>28</sub> × r<sub>29</sub>  
Checksums for d<sub>2</sub>  
d<sub>2</sub>' = r<sub>0</sub> × r<sub>2</sub>  
d<sub>2</sub>' = r<sub>1</sub> × r<sub>30</sub>  
20 d<sub>2</sub>' = r<sub>3</sub> × r<sub>5</sub>  
d<sub>2</sub>' = r<sub>4</sub> × r<sub>6</sub>  
d<sub>2</sub>' = r<sub>7</sub> × r<sub>9</sub>  
d<sub>2</sub>' = r<sub>8</sub> × r<sub>10</sub>  
d<sub>2</sub>' = r<sub>11</sub> × r<sub>13</sub>  
25 d<sub>2</sub>' = r<sub>12</sub> × r<sub>14</sub>  
d<sub>2</sub>' = r<sub>15</sub> × r<sub>17</sub>  
d<sub>2</sub>' = r<sub>16</sub> × r<sub>31</sub>

$$d_2' = r_{18} \times r_{20}$$

$$d_2' = r_{19} \times r_{21}$$

$$d_2' = r_{22} \times r_{24}$$

$$d_2' = r_{23} \times r_{25}$$

5            $d_2' = r_{26} \times r_{28}$

$$d_2' = r_{27} \times r_{29}$$

Checksums for  $d_3$

$$d_3' = r_0 \times r_4$$

$$d_3' = r_1 \times r_5$$

10            $d_3' = r_2 \times r_6$

$$d_3' = r_3 \times r_{30}$$

$$d_3' = r_7 \times r_{11}$$

$$d_3' = r_8 \times r_{12}$$

$$d_3' = r_9 \times r_{13}$$

15            $d_3' = r_{10} \times r_{14}$

$$d_3' = r_{15} \times r_{19}$$

$$d_3' = r_{16} \times r_{20}$$

$$d_3' = r_{17} \times r_{21}$$

$$d_3' = r_{18} \times r_{31}$$

20            $d_3' = r_{22} \times r_{26}$

$$d_3' = r_{23} \times r_{27}$$

$$d_3' = r_{24} \times r_{28}$$

$$d_3' = r_{25} \times r_{29}$$

Checksums for  $d_4$

25            $d_4' = r_0 \times r_8$

$$d_4' = r_1 \times r_9$$

$$d_4' = r_2 \times r_{10}$$

$d_4' = r_3 \times r_{11}$   
 $d_4' = r_4 \times r_{12}$   
 $d_4' = r_5 \times r_{13}$   
 $d_4' = r_6 \times r_{14}$   
5       $d_4' = r_7 \times r_{30}$   
 $d_4' = r_{15} \times r_{23}$   
 $d_4' = r_{16} \times r_{24}$   
 $d_4' = r_{17} \times r_{25}$   
10      $d_4' = r_{18} \times r_{26}$   
 $d_4' = r_{19} \times r_{27}$   
 $d_4' = r_{20} \times r_{28}$   
 $d_4' = r_{21} \times r_{29}$   
 $d_4' = r_{22} \times r_{31}$   
Checksums for  $d_5$   
15      $d_5' = r_0 \times r_{15}$   
 $d_5' = r_1 \times r_{16}$   
 $d_5' = r_2 \times r_{17}$   
 $d_5' = r_3 \times r_{18}$   
 $d_5' = r_4 \times r_{19}$   
20      $d_5' = r_5 \times r_{20}$   
 $d_5' = r_6 \times r_{21}$   
 $d_5' = r_7 \times r_{22}$   
 $d_5' = r_8 \times r_{23}$   
 $d_5' = r_9 \times r_{24}$   
25      $d_5' = r_{10} \times r_{25}$   
 $d_5' = r_{11} \times r_{26}$   
 $d_5' = r_{12} \times r_{27}$

$$d_5' = r_{13} \times r_{28}$$

$$d_5' = r_{14} \times r_{29}$$

$$d_5' = r_{30} \times r_{31}$$

$r_n$  ( $n = 0, 1, \dots, 31$ ) represents the 31-level

5 (31-bit in the case of hard decision) signal supplied  
to the checksum calculator 16 after being multiplied by  
the mask pattern.

In step S20, these 80 outputs in total are decided  
by majority by the majority decision unit 18, and  $d_1'$   
10 to  $d_5'$  are decoded. To be more specific, concerning  
the checksum output, it is decided to be 0 if 0 is  
majority, and 1 if 1 is majority.

In step S22, the orthogonal code multiplier 20  
multiplies 5-bit information data  $d_1'$  to  $d_5'$  by the  
15 orthogonal codes corresponding to the 5-bit information  
data  $d_1'$  to  $d_5'$ . The output from the orthogonal code  
multiplier 20 is as follows.

$$d_1'C_1 \wedge d_2'C_2 \wedge d_3'C_3 \wedge d_4'C_4 \wedge d_5'C_5 \quad (4)$$

In step S24, the exclusive OR circuit 22  
20 calculates the exclusive OR of the output (expression  
(3)) from the exclusive OR circuit 14 and the output  
(expression (4)) from the orthogonal code multiplier  
20. The exclusive OR, which is output from the  
exclusive OR circuit 22 is as follows.

$$\begin{aligned} & d_0C_0 \wedge (d_1 \wedge d_1')C_1 \wedge (d_2 \wedge d_2')C_2 \wedge (d_3 \wedge d_3')C_3 \wedge (d_4 \wedge \\ & d_4')C_4 \wedge (d_5 \wedge d_5')C_5 \wedge (d_6 \wedge d_6')M_1 \wedge (d_7 \wedge d_7')M_2 \wedge (d_8 \wedge d_8')M_3 \\ & \wedge (d_9 \wedge d_9')M_4 \wedge e \end{aligned} \quad (5)$$

Here, if  $d_1'$  to  $d_9'$  are correctly decoded, the term of  $(d_n \wedge d_n')C_n$  ( $n = 1, 2, \dots, 9$ ) becomes a 0 vector. In this case, the output (expression (5)) from the exclusive OR circuit 22 is as follows.

$$5 \quad d_0 C_0 \hat{\rightarrow} e \quad (6)$$

Since  $C_0$  is all 1,  $d_0'$  can be obtained by judging the output (expression (6)) from the exclusive OR circuit 22 by the majority decision unit 24 (step S26). To be more specific, each bit of the information data is decided to be 0 if 0 is majority, and 1 if 1 is majority in the output (expression (6)) from the exclusive OR circuit 22. When bit  $d_0'$  of the information data is determined by the majority decision unit 24, bits  $d_6'$  to  $d_9'$  of the information data can be determined from the mask pattern used for this determination. The operation mentioned above allows to determine respective bits  $d_0'$  to  $d_9'$  of the information data.

20 This information data  $d_0'$  to  $d_9'$  is Reed-Muller  
encoded by the Reed-Muller encoder 26 as follows, in  
step S28.

$$d_0'c_0 \wedge d_1'c_1 \wedge d_2'c_2 \wedge d_3'c_3 \wedge d_4'c_4 \wedge d_5'c_5 \wedge d_6'm_1 \\ d_7'm_2 \wedge d_8'm_3 \wedge d_9'm_4 \quad (7)$$

In step S30, the Euclidean distance calculator 28  
25 calculates the Euclidean distance between the output  
(expression (7)) from the Reed-Muller encoder 26 and  
the received coded signal (expression (2)) output from

the hard decision unit 10. To be more specific, first, the exclusive OR of the output (expression (7)) from the Reed-Muller encoder 26 and the output (expression (2)) from the hard decision unit 10 is obtained as follows:

$$(d_0 \wedge d_0')C_0 \wedge (d_1 \wedge d_1')C_1 \wedge (d_2 \wedge d_2')C_2 \wedge (d_3 \wedge d_3')C_3 \wedge (d_4 \wedge d_4')C_4 \wedge (d_5 \wedge d_5')C_5 \wedge (d_6 \wedge d_6')M_1 \wedge (d_7 \wedge d_7')M_2 \wedge (d_8 \wedge d_8')M_3 \wedge (d_9 \wedge d_9')M_4 \wedge e \quad (8)$$

Expression (8) represents a 32-bit signal, and the sum of these 32 bits represents the Euclidean distance between the output (expression (7)) from the Reed-Muller encoder 26 and the output (expression (2)) from the hard decision unit 10.

In step S32, it is determined whether the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12. If non-processed mask patterns remain, the next mask pattern is designated in step S34, and the readout of mask pattern in step S14 and following processing is repeated.

When the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12, the minimum distance detector 30 detects the minimum Euclidean distance in step S36. The information data  $d_6'$  to  $d_9'$  are decoded based on the mask pattern at the time when the minimum Euclidean distance is detected. The information data  $d_0'$  to  $d_9'$  are

decoded based on the information data  $d_6'$  to  $d_9'$  together with  $d_1'$  to  $d_5'$  decoded by the majority decision unit 18 and  $d_0'$  decoded by the majority decision unit 24,

5 As mentioned above, according to the present embodiment, a processing of Reed-Muller decoding by majority decision with the mask symbols removed from a Reed-Muller code using mask symbols, Reed-Muller coding the sum of this decoding result and the mask symbols,  
10 and calculating the Euclidean distance between this coded output and the original code is repeated for the number of times as the number of mask symbols, mask symbols corresponding to the minimum distance are determined. The information data are decoded by using  
15 these mask symbols. Therefore, the number of checksums to be calculated for the majority decision does not increase compared to the case of Reed-Muller code decoding without using mask symbols. Consequently, a decoding apparatus that can reduce the operation load  
20 and the hardware scale can be supplied.

This embodiment can also be used as decoding apparatus of (32, 6) Reed-Muller code, without limiting to (32, 10) Reed-Muller code. For this purpose, a changeover switch 32 is connected between the hard decision unit 10 and the exclusive OR circuit 14, and provides a path for directly supplying the output from the hard decision unit 10 to the checksum calculator 16

bypassing the exclusive OR circuit 14. A changeover switch 34 is connected also between the majority decision unit 24 and the Reed-Muller encoder 26, and the output of the majority decision unit 24 may be 5 output as it is as decoding result.

In the case of the maximum likelihood decoding, it is necessary to calculate correlations between the coded signal and all the code words. However, the present invention enables to decrease the amount of 10 calculation of the correlations by multiplying the coded signal and the mask symbols beforehand.

FIG. 3 is a modification of the first embodiment in which the checksum calculator 16 and the majority decision unit 18 of FIG. 1 is modified. The modification comprises a memory 40 storing the output from 15 the exclusive OR circuit 14, exclusive OR circuits 42 reading out bit data from the memory 40 and calculating the exclusive ORs, a checksum selector 44 selecting the outputs from the exclusive OR circuits 42 according to the kind of Reed-Muller code, an accumulator 46 accumulatively adding the outputs from the checksum selector 44, and a decision device 48 for hard judging 20 the output from the accumulator 46 and decoding the information bit.

The Reed-Muller code is stored in the memory 40. The combinations of checksums are determined according 25 to the kind of the Reed-Muller code, and exclusive ORs

of the combinations according to this are obtained by the exclusive OR circuit 42. For example, 80 checksums are calculated for (32, 6) Reed-Muller code, while only 32 checksums are calculated for (16, 5) Reed-Muller  
5 code. The outputs from the exclusive OR circuits 42 are selected by the checksum selector 46 for which bit to be used as code, accumulatively added by the accumulator 46, and the bit is decided by the decision device 48.

10 Other embodiments of the decoding apparatus according to the present invention will be described. The same portions as those of the first embodiment will be indicated in the same reference numerals and their detailed description will be omitted.

15 Second embodiment

FIG. 4 shows a second embodiment of the decoding apparatus which is simplifier than the first embodiment.

Comparing the output (expression (5)) of the  
20 exclusive OR circuit 22 and the Euclidean distance  
(expression (8)) between the output (expression (7)) of the Reed-Muller encoder 26 and the output (expression  
(2)) of the hard decision circuit 10, it is found that  
the expression (8) includes  $d_0' C_0$  which is not included  
25 in the expression (5). If  $d_0' = 1$ , the expression (8)  
is an inversion of the expression (5) since  $C_0$  is  
a code of all 1.

Therefore, it can be determined that one of the output (expression (5)) of the exclusive OR circuit 22 and the inverted signal of the output of the exclusive OR circuit 22 which has the shorter Euclidean distance 5 is a correct code. Thus, it is unnecessary to provide the majority decision unit 24, the Reed-Muller encoder 26, and the Euclidean distance calculator 28 of FIG. 1.

A result of accumulation of each bit of the expression (5) represents the Euclidean distance 10 between the output (expression (7)) of the Reed-Muller encoder 26 (where  $d_0' = 0$ ) and the received coded data. A result of accumulation of each bit of an inversion of the expression (5) represents the Euclidean distance 15 between the output (expression (7)) of the Reed-Muller encoder 26 (where  $d_0' = 1$ ) and the received coded data. The number of "1"s included in the accumulation result equals to the Euclidean distance.

Therefore, the output from the exclusive OR circuit 22 is supplied to an inversion detector 54 and 20 the accumulation result of the expression (5) and the accumulation result of an inversion of the expression (5) are compared. Smaller one is supplied to the minimum distance detector 30.

The aforementioned processing is performed for all 25 16 kinds of mask patterns corresponding to  $d_6$  to  $d_9$ , and the minimum Euclidean distance is detected by the minimum distance detector 30. Bit data  $d_0'$  to  $d_9'$

at the time when the minimum distance is detected are considered to be correct, completing the decoding.

This embodiment can also be used as decoding apparatus of (32, 6) Reed-Muller code. Thus, the  
5 changeover switch 32 is connected between the hard decision unit 10 and the exclusive OR circuit 14, and the changeover switch 34 is connected between the inversion detector 54 and the minimum distance detector 30.

10 FIG. 5 is a flow chart of the second embodiment. Step S10 to step S24 of FIG. 5 are the same as those of FIG. 2. In the second embodiment, after step S24 in which the exclusive OR circuit 22 calculates the exclusive OR of the output (expression (3)) from the  
15 exclusive OR circuit 22 and the output (expression (4)) from the orthogonal code multiplier 20, the inversion detector 54 calculates in step S40 the accumulation result of bits of the output from the exclusive OR circuit 22 and the accumulation result of bits of the inverted output from the exclusive OR circuit 22.  
20 In step S42, the smaller one of the two accumulation results is selected and is supplied to the minimum distance detector 30.

25 In step S32, it is determined whether the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12. If non-processed mask patterns remain, the next mask pattern

is designated in step S34, and the readout of mask pattern in step S14 and following processing is repeated.

When the aforementioned processing is performed  
5 for all 16 kinds of mask patterns stored in the memory  
12, the minimum distance detector 30 detects in step  
S36 the minimum Euclidean distance.

### Third embodiment

FIG. 6 shows a decoding apparatus of (32, 10)  
10 Reed-Muller code according to the third embodiment.  
Though the first and second embodiments relate to the  
hard decision, the third embodiment relates to a soft  
decision.

The following definition will be used for the  
15 description below.

" $\wedge$ " means an exclusive OR operation. For two  
vectors, A and B, " $A \wedge B$ " represents the exclusive OR of  
components of respective vectors A and B.

$m(A)$  represents the vector A in which each of  
20 components 0 and 1 is changed to +1 and -1.

10-bit information data to be encoded are assumed  
to be  $d_0, d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8$  and  $d_9$ .  
Each bit data  $d_n$  is 0 or 1.

Orthogonal codes used for encoding are assumed to  
25 be  $C_0, C_1, C_2, C_3, C_4$  and  $C_5$ . Each code  $C_n$  is a 32-bit  
data, and 32 elements thereof are 0 or 1. Note that  $C_0$   
is a series of all 1.

Similarly, assuming mask symbols used for encoding  
be  $M_1$ ,  $M_2$ ,  $M_3$  and  $M_4$ . Each mask symbol  $M_n$  is a 32-bit  
data. The mask patterns  $d_6M_1 \wedge d_7M_2 \wedge d_8M_3 \wedge d_9M_4$ ,  
which are exclusive ORs of the mask symbols and the  
information data, have  $2^4 = 16$  patterns.

An encoding apparatus encodes the aforementioned  
information data "d" based on the orthogonal codes  $C_0$   
to  $C_5$  and the mask symbols  $M_1$  to  $M_4$ , and outputs  
the following 32-bit coded signal  $m(s)$ . Here, the  
orthogonal codes and the mask symbols to be multiplied  
with each bit of the information data are  
predetermined.

$$m(s) = m(d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge d_6M_1 \wedge d_7M_2 \wedge d_8M_3 \wedge d_9M_4) \quad (21)$$

In this embodiment, the following signal in which  
an error "e" due to transfer path or noise is added to  
the 32-bit coded signal  $m(s)$  is input to the decoding  
apparatus of FIG. 6.

$$m(d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge d_6M_1 \wedge d_7M_2 \wedge d_8M_3 \wedge d_9M_4) + E \quad (22)$$

A multiplier 60 multiplies the received coded  
signal by the mask pattern which is represented by +1  
and -1 and read from the memory 12.

The output from the multiplier 60 is supplied to  
the checksum calculator 16 in the same manner as the  
first embodiment. The calculator 16 calculates 16  
checksums for each of 5 bits of  $d_1$  to  $d_5$  (80 checksums

in total), among 10-bit information data  $d_0$  to  $d_9$ .

The majority decision unit 18 decides a majority of the 80 checksums output from the checksum calculator 16 to decode bits  $d_1'$  to  $d_5'$  corresponding to the orthogonal codes  $C_1$  to  $C_5$ . To be more specific, concerning the checksum, it is decided to be 0 if it is positive, and 1 if it is negative.

The orthogonal code multiplier 20 multiplies 5-bit data  $d_1'$  to  $d_5'$  by the orthogonal codes.

A multiplier 62 multiplies the output from the multiplier 60 and the output from the orthogonal code multiplier 20 which is represented by +1 and -1. In the same manner as the first embodiment, the majority decision unit 24 decides a majority of the output from the multiplier 62 to decode bit  $d_0'$ . To be more specific, concerning the output from the multiplier 62, it is decided to be 0 if it is positive, and 1 if it is negative. When the bit  $d_0'$  of the information data is determined by the majority decision unit 24, bits  $d_6'$  to  $d_9'$  of the information data can be determined based on the mask pattern used for determining the bit  $d_0'$ .

Thus, the information data  $d_0'$  to  $d_9'$  are determined. The Reed-Muller encoder 26 encodes the determined information data  $d_0'$  to  $d_9'$ . A correlation calculator 64 calculates a correlation between the received coded signal and the output from the Reed-Muller encoder 26.

The aforementioned processing is performed for all 16 kinds of the mask patterns, and the maximum correlation is detected by a maximum detector 66. Bit data  $d_0'$  to  $d_9'$  at the time when the maximum correlation is detected are considered to be correct, completing the decoding.

FIG. 7 is a flow chart of the third embodiment. One mask pattern is specified in step S60, this specified mask pattern is read out from the memory 12 in step S62, and the multiplier 60 multiplies the received coded signal by the mask pattern.

The memory 12 stores the orthogonal codes  $C_0$  to  $C_5$ , mask symbols  $M_1$  to  $M_4$ , and 16 mask patterns  $d_6M_1$   $d_7M_2$   $d_8M_3$   $d_9M_4$  not shown in Table 1. "i" represents a bit position.

Supposing the mask pattern read out from the memory 12 be  $M' = m(d_6'M_1 \wedge d_7'M_2 \wedge d_8'M_3 \wedge d_9'M_4)$ , the product of the received coded signal and the mask pattern will be as follows.

$$m(d_0C_0 \wedge d_1C_1 \wedge d_2C_2 \wedge d_3C_3 \wedge d_4C_4 \wedge d_5C_5 \wedge (d_6 \wedge d_6')M_1 \wedge (d_7 \wedge d_7')M_2 \wedge (d_8 \wedge d_8')M_3 \wedge (d_9 \wedge d_9')M_4) + E \quad (23)$$

In step S66, the checksum calculator 16 calculates the checksum of the expression (23) output from the multiplier 60. Respectively, 16 checksums are calculated for 5 bits of  $d_1$  to  $d_5$ , in the 10-bit information data of  $d_0$  to  $d_9$ .

In step S68, these 80 outputs in total are decided

by majority by the majority decision unit 18, and  $d_1'$  to  $d_5'$  are decoded. To be more specific, concerning the checksum output, it is decided to be 0 if it is positive, and 1 if it is negative.

5        In step S70, the orthogonal code multiplier 20 multiplies 5-bit information data  $d_1'$  to  $d_5'$  by the orthogonal codes corresponding to the 5-bit information data  $d_1'$  to  $d_5'$ . The output from the orthogonal code multiplier 20 is as follows.

10       $m(d_1' C_1 \wedge d_2' C_2 \wedge d_3' C_3 \wedge d_4' C_4 \wedge d_5' C_5) \quad (24)$

      In step S72, the multiplier 62 multiplies the output (expression (23)) from the multiplier 60 and the output (expression (24)) from the orthogonal code multiplier 20. The output from the multiplier 62 is as follows.

$$m(d_0 C_0 \wedge (d_1 \wedge d_1') C_1 \wedge (d_2 \wedge d_2') C_2 \wedge (d_3 \wedge d_3') C_3 \wedge (d_4 \wedge d_4') C_4 \wedge (d_5 \wedge d_5') C_5 \wedge (d_6 \wedge d_6') M_1 \wedge (d_7 \wedge d_7') M_2 \wedge (d_8 \wedge d_8') M_3 \wedge (d_9 \wedge d_9') M_4) + E \quad (25)$$

20      Here, if  $d_1'$  to  $d_9'$  are correctly decoded, the term of  $(d_n \wedge d_n') C_n$  ( $n = 1, 2, \dots, 9$ ) becomes a 0 vector. In this case, the output (expression (25)) from the multiplier 62 is as follows.

$$m(d_0 C_0) + E \quad (26)$$

25      Since  $C_0$  is all 1,  $d_0'$  can be obtained by judging the output (expression (26)) from the multiplier 62 by the majority decision unit 24 (step S74). To be more specific, each bit of the information data is decided

to be 0 if it is positive, and 1 if it is negative in  
the output (expression (26)) from the multiplier 62.  
When bit  $d_0'$  of the information data is determined by  
the majority decision unit 24, bits  $d_6'$  to  $d_9'$  of the  
5 information data can be determined from the mask  
pattern used for this determination. The operation  
mentioned above allows to determine respective bits  $d_0'$   
to  $d_9'$  of the information data.

This information data  $d_0'$  to  $d_9'$  is Reed-Muller  
10 encoded by the Reed-Muller encoder 26 as follows, in  
step S76.

$$m(d_0'c_0 \wedge d_1'c_1 \wedge d_2'c_2 \wedge d_3'c_3 \wedge d_4'c_4 \wedge d_5'c_5 \wedge d_6'm_1 \wedge \\ d_7'm_2 \wedge d_8'm_3 \wedge d_9'm_4) \quad (27)$$

In step S78, the correlation calculator 64  
15 calculates the correlation between the output  
(expression (27)) from the Reed-Muller encoder 26  
and the received coded signal (expression (22)). To be  
more specific, first, the product of the output  
(expression (27)) from the Reed-Muller encoder 26 and  
20 the received coded signal (expression (22)) is obtained  
as follows:

$$m((d_0 \wedge d_0')c_0 \wedge (d_1 \wedge d_1')c_1 \wedge (d_2 \wedge d_2')c_2 \wedge (d_3 \wedge d_3')c_3 \\ \wedge (d_4 \wedge d_4')c_4 \wedge (d_5 \wedge d_5')c_5 \wedge (d_6 \wedge d_6')m_1 \wedge (d_7 \wedge d_7')m_2 \wedge (d_8 \\ \wedge d_8')m_3 \wedge (d_9 \wedge d_9')m_4) + E \quad (28)$$

25 Expression (28) represents a 32-bit signal, and  
the accumulation result of these 32 bits represents the  
correlation between the output (expression (27)) from

the Reed-Muller encoder 26 and the received coded signal (expression (22)).

In step S80, it is determined whether the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12. If non-processed mask patterns remain, the next mask pattern is designated in step S82, and the readout of mask pattern in step S62 and following processing is repeated.

When the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12, the maximum detector 66 detects the maximum correlation in step S84. The information data  $d_6'$  to  $d_9'$  are decoded based on the mask pattern at the time when the maximum correlation is detected. The information data  $d_0'$  to  $d_9'$  are decoded based on the information data  $d_6'$  to  $d_9'$  together with  $d_1'$  to  $d_5'$  decoded by the majority decision unit 18 and  $d_0'$  decoded by the majority decision unit 24,

As mentioned above, according to the present embodiment, a processing of Reed-Muller decoding by majority decision with the mask symbols removed from a Reed-Muller code using mask symbols, Reed-Muller coding the sum of this decoding result and the mask symbols, and calculating the correlation between this coded output and the original code is repeated for the number of times as the number of mask symbols, mask

symbols corresponding to the maximum correlation are determined. The information data are decoded by using these mask symbols. Therefore, the number of checksums to be calculated for the majority decision does not  
5 increase compared to the case of Reed-Muller code decoding without using mask symbols. Consequently, a decoding apparatus that can reduce the operation load and the hardware scale can be supplied. Further, this embodiment utilizes the soft decision. The majority  
10 decision in the soft decision system is performed at a higher precision than in the hard decision system.

This embodiment can also be used as decoding apparatus of (32, 6) Reed-Muller code. Thus, the changeover switch 32 is connected between the coded  
15 signal input terminal and the multiplier 60, and the changeover switch 34 is connected between the majority decision unit 24 and the Reed-Muller encoder 26.

FIG. 8 is a modification of the third embodiment in which the checksum calculator 16 and the majority decision unit 18 of FIG. 6 is modified. The modification comprises the memory 40 storing the output from the multiplier 60, multipliers 62 reading out bit data from the memory 40 and calculating products, the checksum selector 44 selecting the outputs from the  
20 multipliers 70 according to the kind of Reed-Muller code, the accumulator 46 accumulatively adding the outputs from the checksum selector 44, and the decision  
25

device 48 for hard judging the output from the accumulator 46 and decoding the information bit.

Fourth embodiment

FIG. 9 shows the fourth embodiment of the decoding apparatus which is simplifier than the third embodiment.

Comparing the output (expression (25)) of the multiplier 22 and the correlation (expression (28)) between the output (expression (27)) of the Reed-Muller encoder 26 and the output (expression (22)) of the received coded signal, it is found that the expression (28) includes  $d_0' C_0$  which is not included in the expression (25). If  $d_0' = 1$ , the expression (28) is an inversion of the expression (25) since  $C_0$  is a code of all 1.

Therefore, it can be determined that one of the accumulation result of the output (expression (25)) of the multiplier 62 and the accumulation result of the inverted signal of the output of the multiplier 62 which is larger can be used as the correlation. Thus, it is unnecessary to provide the majority decision unit 24, the Reed-Muller encoder 26, and the correlation calculator 64 of FIG. 6.

A result of accumulation of each bit of the expression (25) equals to the correlation between the output (expression (27)) of the Reed-Muller encoder 26 (where  $d_0' = 0$ ) and the received coded data (expression

(22)). A result of accumulation of each bit of  
an inversion of the expression (25) equals to the  
correlation between the output (expression (27)) of the  
Reed-Muller encoder 26 (where  $d_0' = 1$ ) and the received  
5 coded data.

Therefore, the output from the multiplier 62 is supplied to an inversion detector 78 and the accumulation result of the expression (25) and the accumulation result of an inversion of the expression (25) are  
10 compared. Larger one is supplied to the maximum distance detector 66.

The aforementioned processing is performed for all 16 kinds of mask patterns corresponding to  $d_6$  to  $d_9$ , and the maximum correlation is detected by the maximum  
15 detector 66. Bit data  $d_0'$  to  $d_9'$  at the time when the maximum correlation is detected are considered to be correct, completing the decoding.

This embodiment can also be used as decoding apparatus of (32, 6) Reed-Muller code. Thus, the  
20 changeover switch 32 is connected between the coded signal input terminal and the multiplier 60, and the changeover switch 34 is connected between the inversion detector 78 and the maximum detector 66.

FIG. 10 is a flow chart of the fourth embodiment.  
25 Step S60 to step S72 of FIG. 10 are the same as those of FIG. 7. In the fourth embodiment, after step S72 in which the multiplier 62 calculates the product of the

output (expression (23)) from the multiplier 60 and the output (expression (24)) from the orthogonal code multiplier 20, the inversion detector 78 calculates in step S90 the accumulation result of bits of the output from the multiplier 62 and the accumulation result of bits of the inverted output from the multiplier 62. In step S92, the larger one of the two accumulation results is selected and is supplied to the maximum detector 66.

In step S80, it is determined whether the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12. If non-processed mask patterns remain, the next mask pattern is designated in step S82, and the readout of mask pattern in step S62 and following processing is repeated.

When the aforementioned processing is performed for all 16 kinds of mask patterns stored in the memory 12, the maximum detector 66 detects in step S84 the maximum correlation.

Additional advantages and modifications will readily occur to those skilled in the art. Therefore, the invention in its broader aspects is not limited to the specific details and representative embodiments shown and described herein. Accordingly, various modifications may be made without departing from the spirit or scope of the general inventive concept as

defined by the appended claims and their equivalents.